MANUEL D'UTILISATION DU SCRIPT updateRs05112010.pl

Auteur : Alexandre DUVAL
Mail : alexandre.duval@nantes.fr
date de création : 5 novembre 2010

I)Les données

I.1)Entrées

I.1.1) Les fichiers BIM

Le script a été conçu pour réaliser le traitement d'une série de un ou plusieurs fichiers BIM (format pLink) placé(s) dans le répertoire ./in/bim.

I.1.2) Les fichiers FRQ (Optionnel)

Le programme test automatiquement la présence d'un fichier de fréquences au format plink (Attention le programme ne test pas ce format) de même nom que le fichier BIM et charge les fréquences. Si le fichier est absent, cela n'interrompt pas pour autant l'exécution du programme (caractère facultatif).

Les fichiers FRQ doivent être placés dans le répertoire ./in/frequencies

I.2)Données annexes (Repertoire ./db)

I.*2.1) RsmergeArch.bcp.gz*

Disponible sur le ftp de ncbi :
ftp://ftp.ncbi.nlm.nih.gov/snp/database/organism_data/human_9606/RsMergeArch.bcp.gz
Dernière mise à jour du fichier : le 11 mai 2010

/!\ Fichier tab délimité

oldrs	newrs	build(dbsnp)	orientation	Create time	Update time	Current-rs	Current-orientation
840	715	85		2000-09-19 14:28:00.0	2004-10-10 11:29:00.0	715	0
905	716	85		2000-09-19 14:28:00.0	2004-10-10 11:35:00.0	716	0
1086	940	123	0	2004-10-08 09:45:00.0	2004-10-18 15:43:00.0	940	0
1234	1067	126	0	2006-03-11 07:22:00.0	2006-03-11 07:22:00.0	1067	0

Rsname(old ou new) =concaténation de "rs" et oldrs (ou "rs" et newrs).

Le fichier RsMerge est utilisé pour « merger » l'ensemble des identifiants « rs » qui correspondent au même SNP (même position, même contig mais rsname différent) avant d'effectuer une recherche dans les données 1000génomes et hapmap.

I.*2.2)Données 1000genomes (Repertoire ./db/1000g_jun2010_b36_ceu)*

Disponibles via le site dédié à IMPUTE SOFTWARE de Jonathan MARCHINI
(https://mathgen.stats.ox.ac.uk/impute/impute.html)
via le lien : https://mathgen.stats.ox.ac.uk/impute/1000g_jun2010_b36_ceu.tgz
dernière mise à jours : mars 2010 pour les SNPs (nom et position) et juin 2010 pour les haplotypes.
Population : CEU

2 types de données disponiblent

Pour chacun des 22 autosomes :
- un fichier contenant la liste des SNPs, avec nom position et allèles (*.legend) - fichier espace délimité.
- un fichier haplotypes (*.haps) - fichier espace délimité.

Pour le programme un seul fichier de données espace délimité est constitué :
1000g_jun2010_b36_ceu.fulldata.db en utilisant le script calc_frequence.pl
Par simple comptage des allèles dans le panel le script calcul la fréquence allélique de l'allèle a0.
Cette fréquence est ensuite associé au SNP correspondant.

```
chr rsID position a0 a1 freq_a0
1 1-533 533 G C 0.950
1 1-41342 41342 T A 0.758
1 1-41791 41791 G A 0.958
1 1-44449 44449 T C 0.983
1 rs2462492 44539 C T 0.983
```

I.2.3)Données Hapmap (Repertoire ./db/hapmap3_r2_b36_ceu+tsi_minus1kG)

Sur le même principe que pour les données 1000génomes un seul et unique fichier de données hapmap3_r2_b36_ceu+tsi_minus1kG.fulldata.db est constitué à partir de fichiers disponible via https://mathgen.stats.ox.ac.uk/impute/hapmap3_r2_b36_ceu+tsi_minus1kG.gz

> Nota : Calcul et jointure des fréquences allèliques observés dans le panel avec la liste de snps correspondants
>
> /!\ Fichier espace délimité.
>
> Script : calc_frequence.pl
> 1er argument : répertoire
> 2ème argument "PRE"(nom de fichier).
>
> Exemple : $ perl calc_frequence.pl
> /tmp/duval_tmp/COMPARE1000G/db/hapmap3_r2_b36_ceu+tsi_minus1kG
> hapmap3_r2_b36_ceu+tsi_minus1kG_chr
> "Chomosome 1 done !"
> "Chomosome 2 done !"
> "Chomosome 3 done !"
> °°°

IN : Liste des marqueurs dataname_chr'i'.legend et haplotypes observés dataname_chr'i'.hap pour chaque chromosome (.haps dans la dernière mise à jours des données 1000genome).

OUT : Un seul fichier dataname_fulldata.db (concaténation de toutes les données initialement sous forme d'un fichier par chromosome) en un seul fichier utilisable.

I.2.4) Genetic_mapping (Pour la définition des blocs d'haplotypes).

Pour chaque chromosomes (hormis(X,Y et M) un fichier genetic_map_chr'i'_combined_b36.txt est disponible. Ce fichier contient l'ensemble des taux de recombinaison et la position génétique, à un point donné du chromosomes (i.e position d'un marqueur).
Utilisation : Création de blocs haplotypiques.

source : Les fichiers genetic_mapping sont disponibles avec les autres données (hap et legend) sur le site https://mathgen.stats.ox.ac.uk/impute/impute.html
Dernière mise à jours : juin 2010

/!\ POUR LE CHROMOSOME X, Y et MITOCHONDRIALE

Uniquement disponible via hapmap il est possible de récupérer un fichier par chromosome avec rsname position allèle1 allèle2 frequence_allèle1 directement à partir des fichiers suivant :
source : ftp://ftp.ncbi.nlm.nih.gov/hapmap/frequencies/2010-08_phaseII+III/
- allele_freqs_chrX_CEU_r28_nr.b36_fwd.txt.gz
- allele_freqs_chrY_CEU_r28_nr.b36_fwd.txt.gz
- allele_freqs_chrM_CEU_r28_nr.b36_fwd.txt.gz

Pour la méthode : récupération des colonnes 0(rsname) 2(position) 10(a0) 13(a1) 11(freq a0) dans cet ordre dans un fichier avec inscription du numéro de chromosome dans la 1ère colone(X=23,Y=24,M=26) les fichiers sont alors mergés avec la data contenant les 22 autres chromosomes (utilisation des commandes shell curl, cut, grep, cat,etc..) .

I.3)Sorties

Chacun des fichiers de sortie est défini de la façon qui suit :
> préfixe_nomdu fichierBIM_dateExecution.out
exemple : control_canada_300k_controls.bim_1222010.out

I.3.1) Control .(tab délimité)

Ce fichier associe à la liste des snps soumis au programme (éventuellement mergés sous un autre rsname), un ensemble de trois informations.
-Colone1 : rsname.
-Colonne 2 : Contient une valeur qui code où et comment on à définit une position mise à jours du snp . Par défaut 1 est associé au SNP s'il a été trouvé dans les données 1000génomes à notre disposition.

2 si on a été obligé de passer par les données hapmap car non présent dans 1000genomes.
3 si on a été obligé de passer par dbSnp130 (hg18/build36).
4 Concerne les snps particuliers ou très récents pour lesquels par défaut la position

physique est calculée en fonction des snps qui l'encadre. /!\ Pour permettre cela les snps du fichier d'entré (BIM) doivent être ordonné en fonction de leur position initiale.

-Colonne 3 : Indique si le SNPs courant est présent dans 1000genomes sous un nom non soumis (non rs) de la forme « n°chr-pos » ou par défaut « NA » .
-Colonne 4 : position du SNP aligné avec les données 1000genomes,hapmap ou dbSNP.

I.3.2) Translation_.(tab délimité)

Ce journal recense l'ensemble de nos marqueurs dont le numéro rs a été "mergé" sous un autre rsname. Seul le nouveau nom est retenu pour le traitement hormis les snps "mergés" non identifiable dans 1000genomes mais présent sous leur ancien nom.

Oldname		Newname
rs16825288	devient	rs3737622
rs7368038	devient	rs2643891
rs6424088	devient	rs4648545
rs1763330	devient	rs1081454
rs3747977	devient	rs3205229

I.3.3) frequencies_.(tab délimité)

Ce fichier contient un ensemble d'informations relatif à la liste de snps de la puce.

chrom	rsname	chromEnd	a0	freqDB_a0	freqData_a0	Genetic_Map	block
1	rs3934834	995669	T	0.117	0.1469	0.4963005276	1-2
1	rs3737728	1011278	A	0.275	0.2464	0.5080087000	1-2
1	rs6687776	1020428	T	0.117	0.1722	0.5473720658	1-2
1	rs9651273	1021403	A	0.275	0.255	0.5479372336	1-2
1	rs4970405	1038818	G	0.083	0.1216	0.5529109999	1-2

-Chrom : numero chromosome (1 à 24 et 26 pour le chromosome M)
-rsname : Nom sous forme actualisé (si mergé)
-ChromEnd : position du SNPs (aligné sur les données 1000genomes, hapmap ou dbSNP 130).
-a0 : allèle mineur
-freqDB_a0 : fréquence de l'allèle a0 calculé dans les panel 1000genomes ou hapmap. Pour des question pratique à pour valeur NA si il est impossible de trouver le snps dans 1000genomes ou hapmap.
-freqData_a0 : Report de la fréquence allélique du SNP dans nos données de puce (fichier pLink .frq).
-Genetic_Map :position génétique.
-Bloc haplotypique (chromosome-numBloc, chromosome-HS si positionner sur un hotspot).

NOTA : L'étendu des données genetic_map ne couvrant pas la totalité des marqueurs 1000genomes des données disponibles (motif inconnu). Certain SNP n'ont pu être attribué à un bloc haplotypique donné. Pour une utilisation de ce fichier dans R les valeurs sont donc remplacées par « NA ».

I.3.4) Newbimfile_.(tab délimité)

Dans le même format que le fichier BIM courant, le nouveau bimfile tient compte des nouveaux numero rs attribués si le marqueur a été "mergé" et des positions alignés sur 1000genomes ou hapmap. Le 1er allèle reste l'allèle mineur.

I.3.5) Warnstrand_.(tab délimité)

Lorsque une anomalie de STRAND est détecté entre nos données de puce et les données annexes, alors une ligne est inscrite dans ce fichier.

2 types d'anomalie déclenche une telle inscription.

Type données	chrom	rsname	Allèle mineur	Allèle majeur	(a0 a1) dans 1kG ou hapmap

1	1	rs624329	G	A	(C T)

Strand Différent.

2	24	rs4933045	A	G	(C T)

Dans un premier temps le strand diffère entre les données de puce et les données hapmaps pour le SNP rs933045, puis l'allèle mineur (BIM file) n'est pas l'allèle de référence dans hapmap.

Autre anomalie pouvant être détecté.

2	24	rsXXXX	A	G	(A C)

Ce genre d'anomalie ne fait l'objet d'aucun traitement particulier, hormis l'inscription d'une alerte dans ce fichier et l'ajout du terme « NA » dans le champ freqDB_a0 du fichier frequencies_.

I.3.6) Missfile1000g_.(tab délimité)

Se fichier contient la liste des marqueurs pour lesquelles une position par défaut (en fonction des marqueurs encadrant) a été attribué.

Parmi les causes : on peut citer le caractère très récent du marqueur. Pour un tel marqueur, étant donner que les données 1000g et hapmap à notre disposition sont alignées sur l'assemblage génomique NCBI36/UCSC hg18. Le script ne peut pas pour le moment ciblé une position pour des marqueurs au pire absent dans dbSNP130(hg18).

I.3.7) Double1000g_.(tab délimité)

A l'origine créer pour identifier les marqueurs référencés plusieurs fois dans 1000g (doublons avec même rsname, même allèle mais position différente) . Néanmoins la dernière release des données montre que les redondances ont été corrigés. Ce fichier est généré à titre de simple contrôle.

II)Le Script

Le programme UpdateRsname se résume à 3 élément :

-Un script principal de lancement en perl : updateRs05112010.pl .
-2 librairie de fonction perl et perl DBI.

1) UpdateRsSub.pm : contient les fonctions de traitement des données du script.
2) UpdteRsUtils.pm : contient une série de fonctions utilitaires (toute ne sont pas nécessaire), permettant la lecture du fichier de configuration, lecture du fichier RsMerge sous format compressé GZIP, la création de sous répertoire « report » si il a été supprimer...et d'autre fonctions.

Les Fonctions

1)Script de configuration

1.a) Le fichier de Configuration

Sous un format correspondant aux fichiers de configuration windows (.ini). Les paramètres de navigation et d'exécution nécécessaire au script sont entrés dans ce fichier sous la forme : paramètre=valeur. Les différents ensembles de paramètres sont regroupés en section [SECTION] afin de clarifier la lecture et l'écriture (modifications) par l'utilisateur.

```
#commentaires
[SECTION1]
paramètre1=valeurA
paramètre2=valeurB

[SECTION2]
paramètre1=valeurC
paramètre2=valeurD
```

1.b) La fonction READ_CONFIG_INIFILE.

Disponible via la librairie updteRsUtils.pm cette subroutine permet de charger les paramètres du script dans un tableau de référence (table de hash).

2 types de paramètres :

1) Les paramètres de navigation qui permettent au script de localiser les fichiers nécessaires.
2) Autre paramètre : assemblage génomique des données 1000genomes (optionnel).
3) Valeur seuil du taux de recombinaison pour définir les blocs

d'haplotypes. (Recombination_Rate).

2) UPDATERSNAME

Fonction principale du script.

Les Arguments

E -$PATH : PATH du répertoire générale où ce situe l'application et les données
E -$dbRef : Paramètre de configuration relative à la section Database. Contient les paths, et noms des fichiers de données 1000génomes et hapmap.
E -$buildref : (Facultatif) Permet de stopper l'application si l'assemblage génomique des données n'est pas conforme aux attentes de l'utilisateur (alignement sur hg18 comme 1000g). Cet argument n'est plus utilisé.
E -$pathdata : Nom du repertoire qui contient l'ensemble des fichiers BIM que l'on souhaite traiter
E -$pathfreq : Nom du repertoire qui contient l'ensemble des fichiers FRQ associés.
E -$filename : nom du fichier courant.
E -$out : Nom du repertoire où seront accessibles les fichiers de sorties.
E -$date : date du jour (la date est inscrite dans le nom de chaque fichier de sortie au format mmddyyyy)

Fonctionnement

a) Chargement des données annexes.

-RsMergeArch.bcp → LOAD_RsMERGE($,\%,\%)
-Fichier Fréquence correspondant au BIM courant s'il existe → FREQ_DATA($,\%)
-Définition des blocs haplotypiques en fonction du seuil déterminer par l'utilisateur (Recombination_Rate=5,section Database dans le fichier de configuration). Lancer 2 fois à partir des données 1000genomes puis hapmap → UPDATE_RM($,$,$,$,$) ;
-Chargement des données 1000genomes et Hapmap → LOAD_DB(\%, $,$,$) ;

b) Connection à dbSNP

-Connection via perl DBI à la base de données local contenant dbSNP125(hg17), dbSNP130(hg18) et dbSNP131(hg19).
-Préparation des « statements » de requêtes MYSQL.

c) Détermination de l'assemblage génomique correspondant à nos données.

-CONTROL_BUILD($,$,$,$) ;

d) Recherche, mise à jours et aligneement des positions avec 1000genomes, hapmap ou dbSNP130.

-COMPARE1000G(\%,\%,\%,\%,$,\%,$,$,$,$,$,\%,\$)

e) Control si parmis nos données, des SNPs ne sont pas redondant dans 1000genomes.

-CONTROL_DUPLICATE_1000G(\%,$,$,$,$)

3) LOAD_RsMERGE($,\%,\%)

Chargement d'une miniDatabase pour mise à jours des numero Rs.

Arguments

E $RsMergeFile : Nom (en full path) du fichier RsMergeAch.bcp
E/S $newNameRef : Table de hash où chaque rs mergé (clé) est associé le nouveau nom (valeur).
E/S $existMergeRef : Booléan par tableau associatif. Prend 1 si la clé existe dans Rsmerge.

4) FREQ_DATA($,\%)

Chargement des fréquences alléliques associées à nos marqueurs.

Arguments

E $file : Nom du fichier de fréquences correpondant au BIM en cours de traitement (reconnaissance par nom)

E/S $hashRef : Table de hash dans laquelle sont chargés les fréquences.

Fonctionnement

Le fichier FRQ fournit la fréquence allélique de l'allèle mineur (MAF) calculé par plink à partir de nos données de puce. Après simple calcul de la fréquence de l'autre allèle (1-MAF) l'ensemble des fréquences est référencé dans la table avec pour clé la combinaison (rsname,allèle) pour garantir que la bonne fréquence est associé au bon allèle au cour du traitement.

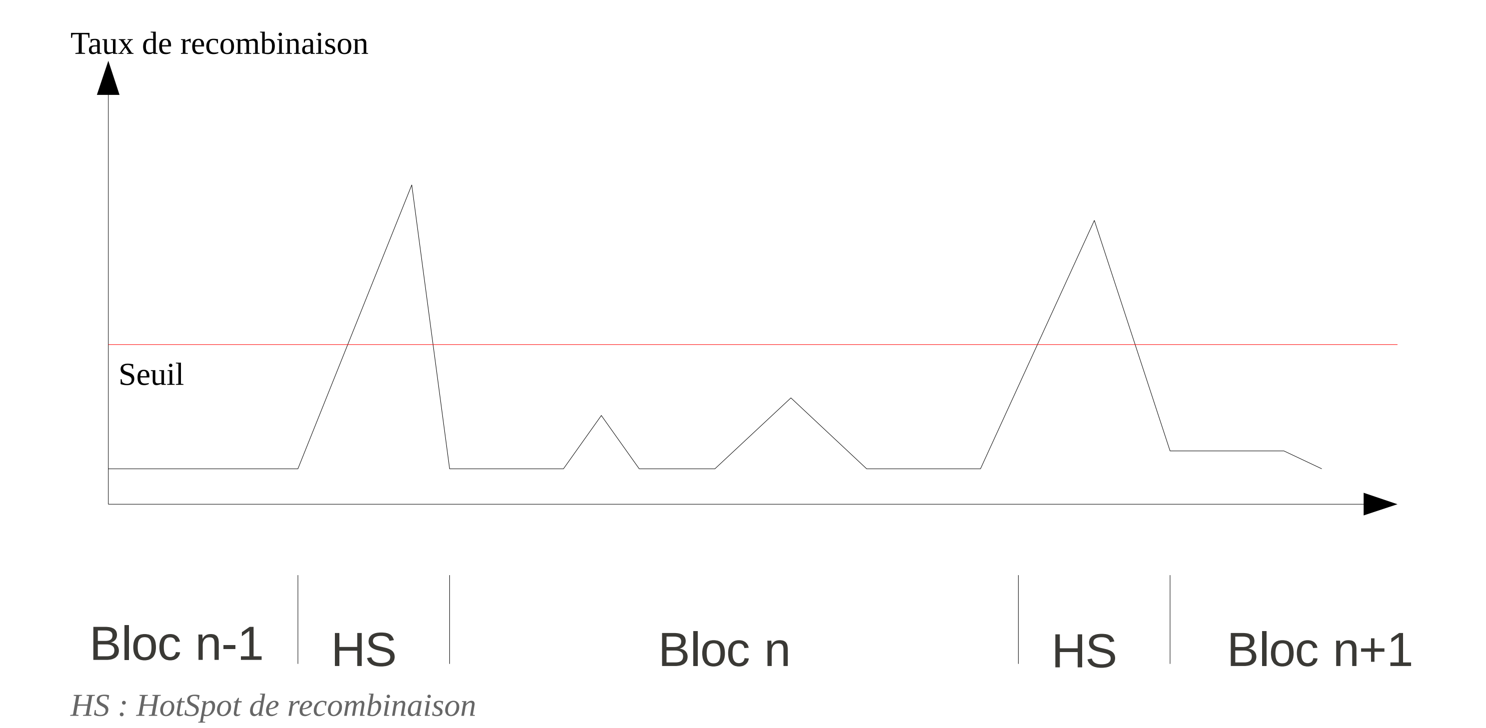
Exemple
```
%$hashRef=(  rs6675798,C=>0.06887
             rs752447,G=>0.02514
             rs6685064,T=>0.083 )
```

5) UPDATE_RM($,$,$,$,$)

Définit le bloc haplotypique en fonction du seuil de recombinaison choisit par l'utilisateur.

Arguments

E -$path : Chemin donnant accès au fichier genetique map
E -$pre : Partie du nom de fichier qui précèdent le numero de chromosomes dans le nom de fichier
E -$post : Partie du nom de fichier qui suit le numero de chromosomes dans le nom de fichier
E -$threeshold : Taux de Recombinaison seuil choisit par l'utilisateur pour caractèrisé le passage d'un bloc halotypique à un hotspot et d'un hotspot au bloc suivant.
S -$outfile : Fichier de sortie, correspondant à une forme concaténé des 22 fichier d'entrés avec l'information de bloc dans la dernière colonne.



HS : HotSpot de recombinaison

Fichier Temporaire : genetic_map_1000G_combined_b36.txt.annot (ou genetic_map_hapmap_combined_b36.txt.annot).
Source ./db/1000g_jun2010_b36_ceu/

<u>->Extrait avec un Seuil à 5.</u>

chromosome position COMBINED_rate(cM/Mb) Genetic_Map(cM) block_partition

1 711153 2.6858076690 0 1-1
1 713682 2.8222713027 0.0067924076 1-1
1 713754 2.9813105581 0.0069956111 1-1
1 718105 2.9806151254 0.0199672934 1-1
...

1 782343 4.4923190939 0.1364244328 1-1
1 787889 5.7728684628 0.1613388344 1-HS
1 788664 5.7751919537 0.1658128075 1-HS
1 788822 5.4012795058 0.1667252878 1-HS
1 789326 5.3521292533 0.1694475327 1-HS
1 789870 5.2371543059 0.1723590910 1-HS
1 798175 4.9596712094 0.2158536575 1-2
1 836727 -1 0.4070589020 1-2
1 871896 -1 0.4070589020 1-2
1 939471 2.0585596818 0.4070589020 1-2
1 952073 1.8166948714 0.4330008711 1-2

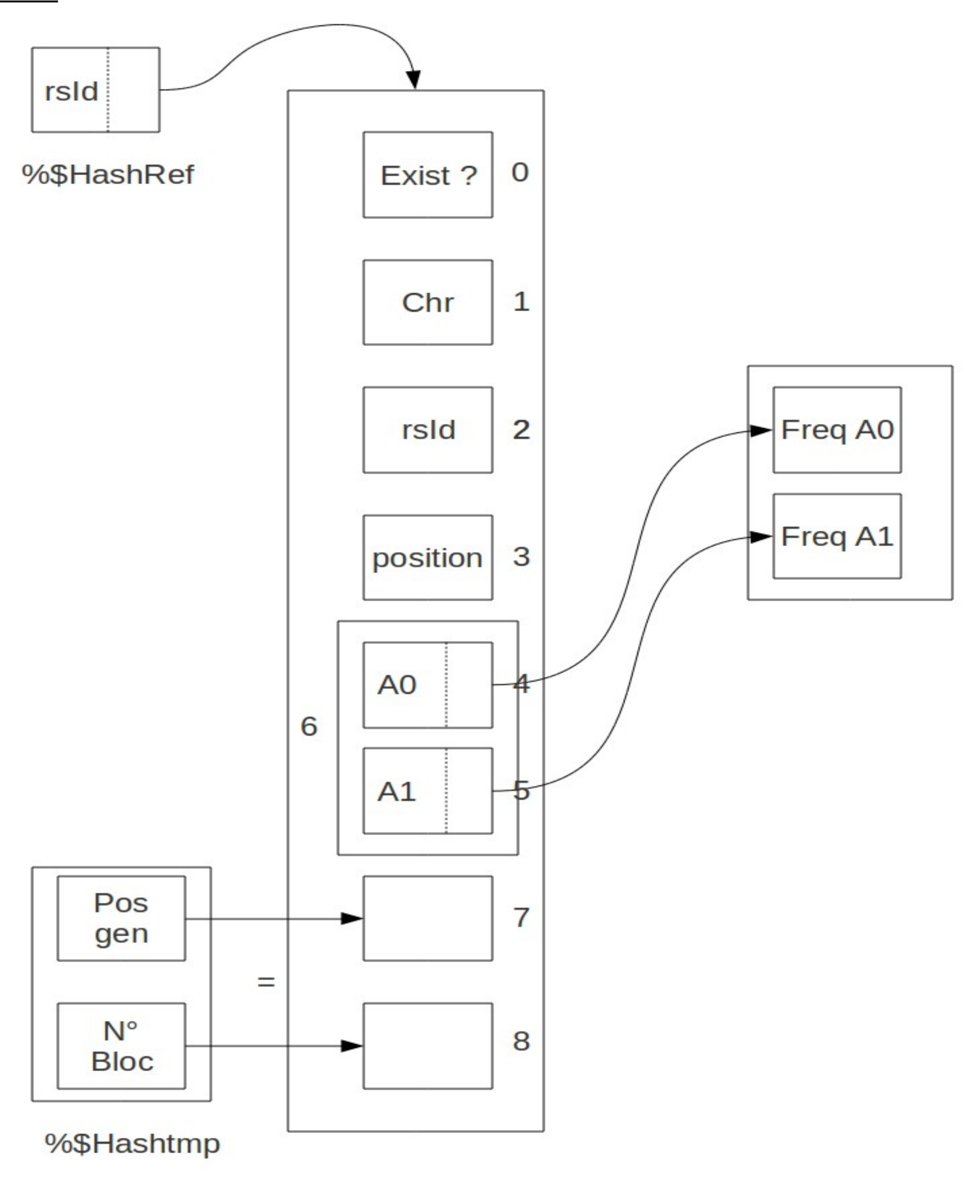
6) LOAD_DB(\%, $,$,$)

Charge l'ensemble des données 1000genomes et hapmap (chr, position, allèle, fréquence allèlique, position géntique, numero de bloc) dans une structure type table de hash.

Arguments

E/S	$HashRef : Table de hash
E	$dir : path du repertoire de données
E	$file1 : fichier de données relative à la position géntique et au numéro de bloc
E	$file2 : fichier de données relative à la position, le rsname, allèle, et fréquence.

Fonctionnement



Loin d'être optimale, cette méthode (quick and Dirty) en perl permet de mimer le principe des bases de données objet (java ou C++). 2 inconvénients : Le temps de chargement est très long et la quantité de mémoire vive utilisées est conséquente mais le temps d'exécution de la partie traitement est rapide.

7) CONTROL_BUILD($,$,$,$)

Permet de consulter par commande DBI,mySQL un ensemble de 3 tables de dbSNP(125,130 et 131), et d'en déduire au mieux l'assemblage génomique des données (fichier BIM).

Arguments :

E	$datafile : Fichier courant (BIM)
E	$dbSNPhg17_SNPpos : Statement de requête MySQL sur la table dbSNP125(database Hg17)
E	$dbSNPhg18_SNPpos : Statement de requête MySQL sur la table dbSNP130(database Hg18)
E	$dbSNPhg19_SNPpos : Statement de requête MySQL sur la table dbSNP131(database Hg19)

Fonctionnement

Après une sélection aléatoire de 500 de nos marqueurs. Une series de requêtes est envoyé au moteur SQL :
Pour chacune des 3 tables reposant sur un assemblage génomique différent une requête de type SELECT position FROM database.table WHERE name=rsname est exécutée.
Les trois requêtes sont renouvelées pour chaque rsname de notre liste. La position retournée pour chacune des trois requêtes est comparée avec la position du marqueur dans nos données. Toute différence stricte incrémente alors un compteur d'erreur suivant le type de requête.

Les compteurs sont analysés à postérieuri et permettent de déduire la build UCSC sur laquelle est axé notre data. L'analyse repose sur trois conditionnelles retenant comme build compatible la build ayant un taux d'erreurs de moins de 10 %.

8) COMPARE1000G

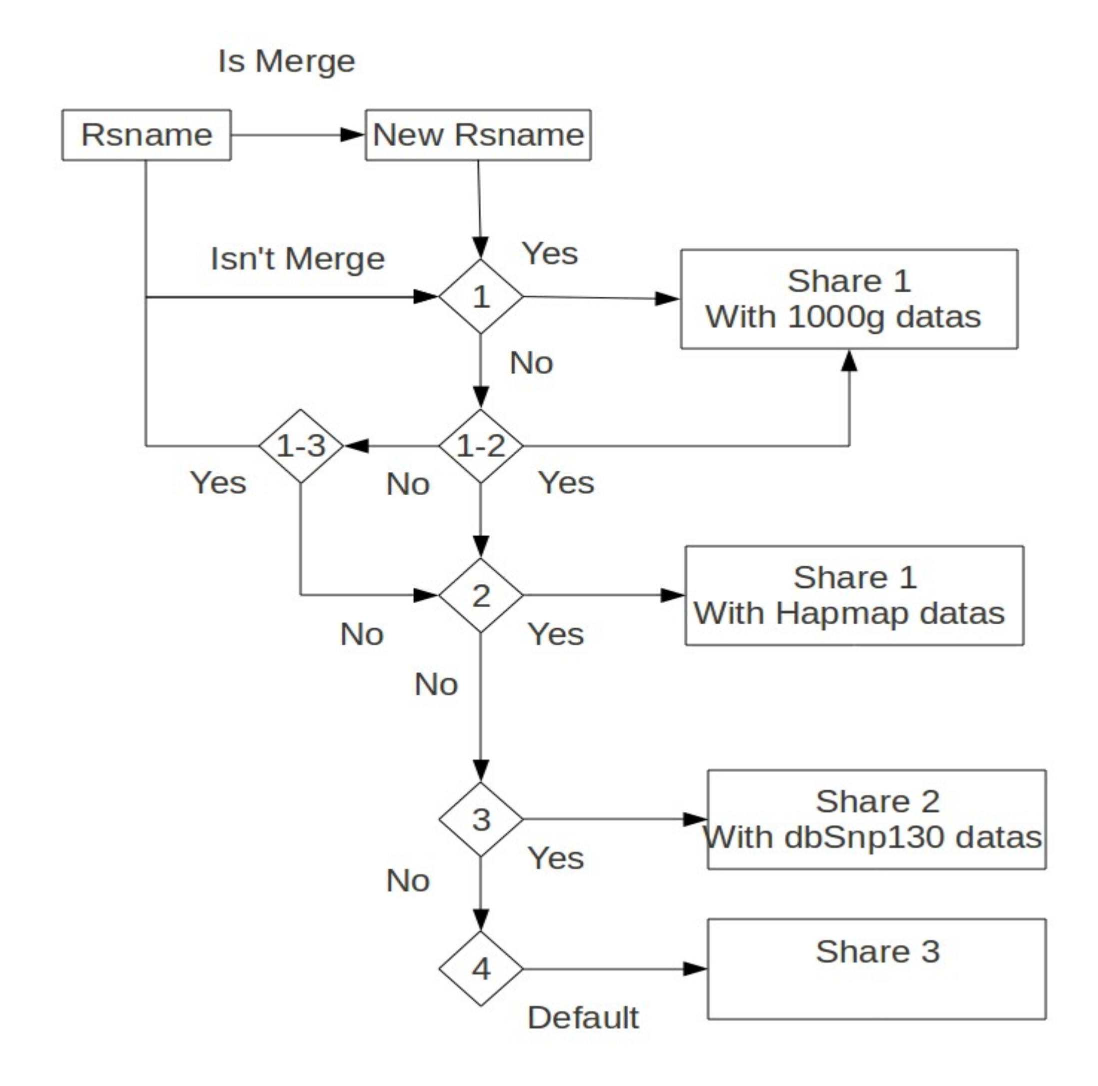
Fonction de traitement qui va chercher a axé les positions de nos marqueurs sur les données panel 1000genomes ou Hapmap (ou alors sur db SNP130). Cette fonction génère également l'ensemble des fichiers de sortie pour chaque bim traité.

Arguments

E	$merge : Table de hash contenant les variable booléenne indiquant si le SNP courant est annoté ou non de le fichier Rsmerge.
E	$new : Table de hash contenant les numeros rs mergé référencé par nom initiale.
E	$kG : Table de hash contenant les données 1000genomes.
E	$hapmap : Table de hash contenant les données hapmap.
E	$dbSNPhg18_SNPrs : Statement DBI retournant position et chr d'un SNP introuvable dans 1000genomes ou hapamap depuis dbSNP130.
E	$freq : Tables des fréquences allèliques.
E	$path : Path des fichier BIM.
E	$path_report : Path du repertoire ou seront écrit les fichier de sortie.
E	$path_db : Path du répertoire des données 1000genomes et hapmap.
E	$file : nom du BIM courant.
E	$Inf : Concaténation du nom de BIM et date d'exécution pour inscription dans les nom des fichiers en sortie.
S	$listSnpRef : Retourne la liste de nos snp pour controler s'il n'y a pas de doublons éventuel dans 1000genomes.

Fonctionnement

A) Principe général



1 Exist in 1000genomes datas
   1-2 Exist in 1000genomes under a alternative name : Chr-position
   1-3 Exist in 1000genomes with a name not Merge

2 Exist in hapmap datas

3 Exist in dbSnp130(Hg18) datas

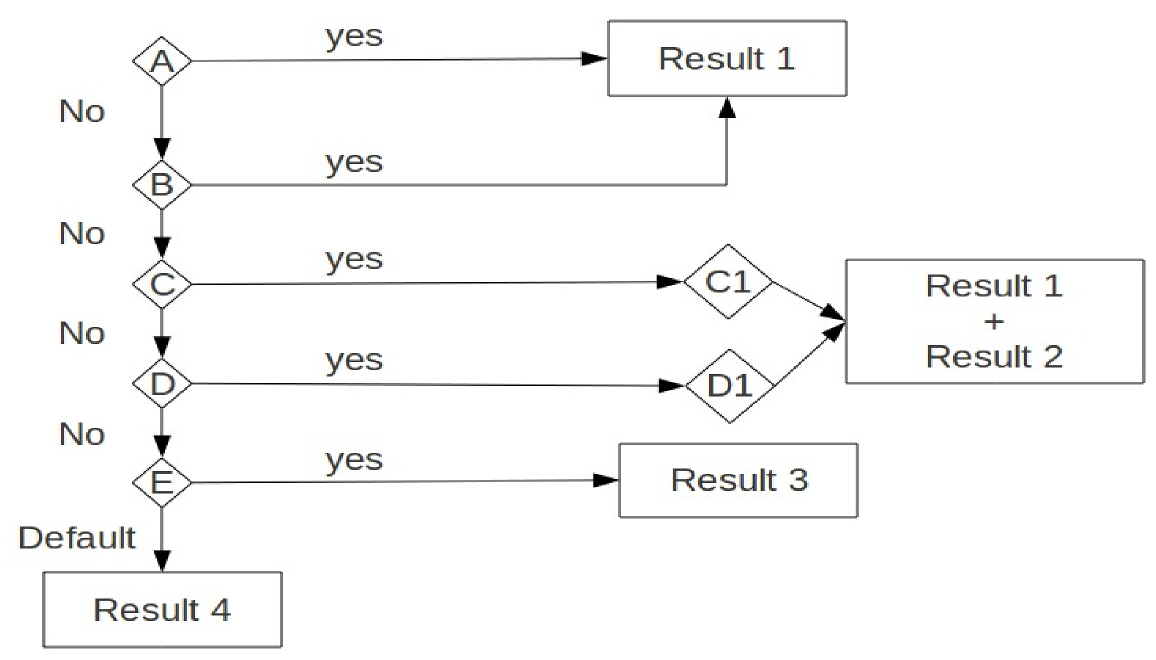
B) Actions

Suivant chacune des conditions à laquelle un SNP à été soumis, un traitement particulier est réalisé.

a) Action 1 (Avec données 1000genomes et Hapmap)



PosPhyData(Current Rsname)=PosPhy1000genome(current Rsname)
ChrData(Current Rsname)=chr1000genome(current Rsname)
PosGenData(Current Rsname)=Pos1000genome(current Rsname)



Result 1
- FreqAll0Database(Current Rsname)=Freq1000genome(current Rsname,All0)
-Print freqFile

Result 2
-Print WarnFile

Result 3
- Default, we define All0 like allele not deleted
- FreqAll0Database(Current Rsname)=Freq1000genome(current Rsname,All0)
-Print freqFile

Result 4
- FreqAll0Database(Current Rsname)= "NA"
-Print freqFile.
-Print WarnFile.

Les Conditionnelles (A,B,C,D,E)

A = L'allèle mineur (mydata) correspond à l'allèle A0 des datas 1000genome ou hapmap et l'allèle majeur correspond à l'allèle A1.

B = Inversement dans la présentation des allèle dans les databases. L'allèle mineur de nos données correspond à l'allèle A1 dans la base et vis et versa. Le référencement par double clé (rsname,allèle) permet d'associé la bonne fréquence.

C = Strand différent entre nos données et les données panels. Une simple conversion par la fonction STRAND permet de retrouver la fréquences de l'alléle A0 dans le panel de référence(C1).

D = Strand différent et inversion. Une simple conversion avec STRAND n'est pas suffisante pour associé la fréquence allèlique dans le panels à all1 du SNP en cours de traitement car l'allèle ne correspond pas au premier allèle dans la base. On test alors s'il correspond aux deuxième allèle et on recupère la fréquence si le test retourne vrai (D1).

E= Un des allèle est délété (-/A ou A/-), on compare alors l'allèle génotypé avec les 2 allèles des données panel (variant ou ancestrale) et on retourne la fréquence correspondantes.

F= Une anomalie autre ne permet pas de trouver une fréquence compatible avec les données panel.

b) Action 2

Envoie de la requête MySQL suivante via DBI:

SELECT chromosome,position FROM hg18.SNP130 WHERE nom=rsname.

Si la commande retourne un tuple alors la position,et le numéro de chromosome retourné sont affectés au SNP sinon on flag est déposé pour passer à l'action 3 au tour suivant. Aucunes fréquences issue d'un panel ou de définition de bloc ne peut être attribué au SNP via dbSNP. Par défaut les valeurs "NA" sont inscrites dans les champs non remplis.

c) Action 3

Si toutes actions sont veine pour pouvoir aligner un SNP. L'action de la dernière chance permet par défaut d'axer d'une certaine façon le SNP sur les données 1000genomes. A chaque lecture de ligne le rsname du précédant SNP lu est stocké dans la variable $previousRS en fin de traitement et sa position est inscrite dans la table $previousPos{$rsname}. Ces 2 données peuvent être alors recupérer à la lecture du SNP suivant. Lorsque une action de la dernière chance est déclenché. Alors une serie d'information est stoké dans la table %Warnmiss :

- Un flag
- Le numero RS du SNP problèmatique.
- a0 et a1.
- Frequence de a0 dans nos datas.
- Le numero RS du SNP précédant.
- La position du SNP précédant.

et $previousRS <- $rsname.

A la ligne suivante, si le flag $warnMiss{$previousRS}[0] est lu à vrai (i.e egale 1) au tour précédant alors un traitement particulier est effectué. Le SNP courant constituant le suivant par rapport au SNP pour lequel on a pas trouvé de position. La position de ce dernier(SNP N-1) est déterminé comme étant la position médiane entre le SNP courant N (à la position $pos) et le SNP N-2 (à la position $warnMiss{$previousRS}[3]).

NOTA : Cette action nécessite de contitionner toute impression dans le newBimfile. Pour éviter toute redondance (2 lignes pour le même rs dont une erronée) une variable booléenne a été ajouté.

III) Annexes

Fichier de configuration : ConfigUpdateRs.ini

```
#[UPDATE RS NAME - parameters file]   #This is a comment

[GENERAL]
path=/tmp/duval_tmp/COMPARE1000G

[DATABASE]

path=/tmp/duval_tmp/COMPARE1000G/db
_1kg=1000g_jun2010_b36_ceu/1000g_jun2010_b36_ceu.fulldata.db
Hapmap=hapmap3_r2_b36_ceu+tsi_minus1kG/hapmap3_r2_b36_ceu+tsi_minus1kG_fulldata.db
Path1kg=/tmp/duval_tmp/COMPARE1000G/db/1000g_jun2010_b36_ceu
PathHapmap=/tmp/duval_tmp/COMPARE1000G/db/hapmap3_r2_b36_ceu+tsi_minus1kG
RecMap_PRE=genetic_map_
RecMap_POST=_combined_b36.txt
Recombination_Rate=5

#build UCSC (hg<?>)
build=18

[IN]
pathBim=/tmp/duval_tmp/COMPARE1000G/in/bim
pathFreq=/tmp/duval_tmp/COMPARE1000G/in/frequencies

[OUT]
path=/tmp/duval_tmp/COMPARE1000G/out
```

*/tmp/duval_tmp désigne la racine de l'application et doit être modifié lorsqu'on souhaite déplacer l'application en l'absence de makefile.